

Computing the Fréchet Distance Between Folded Polygons¹

Atlas F. Cook IV², Anne Driemel³, Sarel Har-Peled⁴, Jessica Sherette⁵, Carola Wenk⁵

Abstract. We present the first results showing that the Fréchet distance between non-flat surfaces can be approximated within a constant factor in polynomial time. Computing the Fréchet distance for surfaces is a surprisingly hard problem. It is not known whether it is computable, it has been shown to be NP-hard, and the only known algorithm computes the Fréchet distance for flat surfaces (Buchin et al.). We adapt this algorithm to create one for computing the Fréchet distance for a class of surfaces which we call folded polygons. Unfortunately, if extended directly the original algorithm no longer guarantees that a homeomorphism exists between the surfaces. We present three different methods to address this problem. The first of which is a fixed-parameter tractable algorithm. The second is a polynomial-time approximation algorithm which approximates the optimum mapping. Finally, we present a restricted class of folded polygons for which we can compute the Fréchet distance in polynomial time.

Keywords: Computational Geometry, Shape Matching, Fréchet Distance

1 Introduction

The *Fréchet distance* is a similarity metric for continuous shapes such as curves and surfaces. In the case of computing it between two (directed open) curves there is an intuitive explanation of the Fréchet distance. Suppose a man walks along one curve, a dog walks along the other, and they are connected by a leash. They can vary their relative speeds but cannot move backwards. Such a walk pairs every point on one curve to one and only one point on the other curve (i.e., creates homeomorphism between surfaces) in a continuous way. The Fréchet distance of the curves is the minimum leash length required for the man and dog to walk along these curves. Although less intuitive, the idea is similar for surfaces.

While the Fréchet distance between polygonal curves can be computed in polynomial time [Alt95], computing it between surfaces is much harder. In

¹ This work has been supported by the National Science Foundation grant NSF CAREER CCF-0643597.

² Department of Mathematics and Computer Science; Eindhoven University of Technology; The Netherlands; a.f.cook@tue.nl

³ Department of Information and Computing Sciences; Utrecht University; The Netherlands; driemel@cs.uu.nl

⁴ Department of Computer Science; University of Illinois; 201 N. Goodwin Avenue; Urbana, IL, 61801, USA; sariel@uiuc.edu

⁵ Department of Computer Science; University of Texas at San Antonio; {jsherett, carola}@cs.utsa.edu

[God98] it was shown that even computing the Fréchet distance between a triangle and a self-intersecting surface is NP-hard. This result was extended in [Buc10] to show that computing the Fréchet distance between 2d terrains as well as between polygons with holes is also NP-hard. Furthermore, while in [Alt10] it was shown to be upper semi-computable, it remains an open question whether the Fréchet distance between general surfaces is even computable.

On the other hand, in [Buc06] a polynomial time algorithm is given for computing the Fréchet distance between two (flat) simple polygons. This was the first paper to give any algorithm for computing the Fréchet distance for a non-trivial class of surfaces and remains the only known approach for computing it. The main idea of their algorithm is to restrict the kinds of different mappings that need to be considered. Our contribution is to generalize their algorithm to a class of non-flat surfaces we refer to as folded polygons. Given that theirs is the only known approach it is of particular importance to explore extending it to new classes of surfaces. The major problem we encountered in generalizing the work of [Buc06] was that the kinds of different mappings which need to be considered is less restricted. We address three different methods to resolve this problem. In Section 3, we outline a fixed-parameter tractable algorithm. In Section 4, we describe a polynomial-time approximation algorithm to compute the Fréchet distance between folded polygons within a constant factor. In Section 5, we describe a nontrivial class of folded polygons for which the original algorithm presented in [Buc06] will compute an exact result.

2 Preliminaries

The Fréchet distance is defined for two k -dimensional hypersurfaces $P, Q : [0, 1]^x \rightarrow \mathbb{R}^d$, where $x \leq d$, as

$$\delta_F(P, Q) = \inf_{\sigma: A \rightarrow B} \sup_{p \in A} \|P(p) - Q(\sigma(p))\|$$

where σ ranges over orientation-preserving homeomorphisms that map each point $p \in P$ to an image point $q = \sigma(p) \in Q$. Lastly, $\|\cdot\|$ is the Euclidean norm but other metrics could be used instead.

Let $P, Q : [0, 1]^2 \rightarrow \mathbb{R}^d$ be connected polyhedral surfaces for each of which we have a convex subdivision. We assume that the dual graphs of the convex subdivisions are acyclic, which means that the subdivisions do not have any interior vertices. We will refer to surfaces of this type as *folded polygons*. We refer to the interior convex subdivision edges of P and Q as *diagonals* and *edges* respectively. Let m and n be the complexities of P and Q respectively. Let k and l be the number of diagonals and edges respectively. Assume without loss of generality the number of diagonals is smaller than the number of edges. Let $T_{matrixmult}(N)$ denote the time to multiply two $N \times N$ matrices.

2.1 Simple Polygons Algorithm Summary

In previous work Buchin et al. [Buc06] compute the Fréchet distance between simple polygons P, Q . The authors show that, while the Fréchet distance between a convex polygon and a simple polygon is the Fréchet distance of their

boundaries, this is not the case for two simple polygons. The idea of their algorithm is to find a convex subdivision of P and map each of the convex regions of it continuously to distinct parts of Q such that taken together the images account for all of Q .

First, they show that the decision problem $\delta_F(P, Q) \leq \varepsilon$ can be solved by (1) mapping the boundary of P , which we denote by ∂P , onto the boundary of Q , which we denote by ∂Q , such that $\delta_F(\partial P, \partial Q) \leq \varepsilon$ and (2) mapping each diagonal d in the convex subdivision of P to a shortest path $f \subseteq Q$ such that both endpoints of f lie on ∂Q and such that $\delta_F(d, f) \leq \varepsilon$.

In order to solve subproblem (1) they use the notion of a free space diagram. For open curves $f, g : [0, 1] \rightarrow \mathbb{R}^d$ it is defined as $FS_\varepsilon(f, g) = \{(x, y) \mid x \in f, y \in g, \|x - y\| \leq \varepsilon\}$ where $\varepsilon \geq 0$. A monotone path starting at the bottom left corner of the free space diagram going to the top right exists if and only if the curves are within Fréchet distance ε . As shown in [Alt95], this can be extended to closed curves by concatenating two copies of the free space diagram to create a *double free space diagram* and searching for a monotone path which covers every point in P exactly once see Figure 1. This algorithm can be used to show whether $\delta_F(\partial P, \partial Q) \leq \varepsilon$ and find the particular mapping(s) between ∂P and ∂Q . In turn this defines a *placement* of the diagonals, i.e., a mapping of the endpoints of the diagonals to endpoints of the corresponding image curves in Q .

Subproblem (2) is solved by only considering paths through the free space diagram that map a diagonal d onto an image curve f such that $\delta_F(d, f) \leq \varepsilon$. Naturally the particular placement of the diagonals determined in subproblem (1) could affect whether this is true. Therefore, they must check this for many paths in the free space diagram. Fortunately they can show that it is sufficient to only consider mapping a diagonal to an image curve which is the shortest path between the end points determined by the placement.

Solving these subproblems generates a mapping between P and Q for ε . This mapping might not be a homeomorphism but the authors show by making very small perturbations of image curves the mapping can be made into one. These perturbations can be arbitrarily small. Thus, because the Fréchet distance is the infimum of all homeomorphisms on the surfaces, the Fréchet distance is ε . For simplicity we will refer to these generated mappings as homeomorphisms. By performing a binary search on a set of critical values they can use the above algorithm for the decision problem to compute the Fréchet distance of P and Q .

2.2 Shortest Path Edge Sequences

Our algorithm extends the simple polygons algorithm to one for folded polygons. The idea of the algorithm is to subdivide one surface, P , into convex regions and pair those with corresponding regions in the other surface, Q . The difference is that those regions of Q are now folded polygons rather than just simple polygons. The authors of [Buc06] show that the Fréchet distance of a convex polygon and a simple polygon is just the Fréchet distance of their boundaries. Using almost the same argument we prove that it also holds for folded polygons, see Appendix A for the full proof

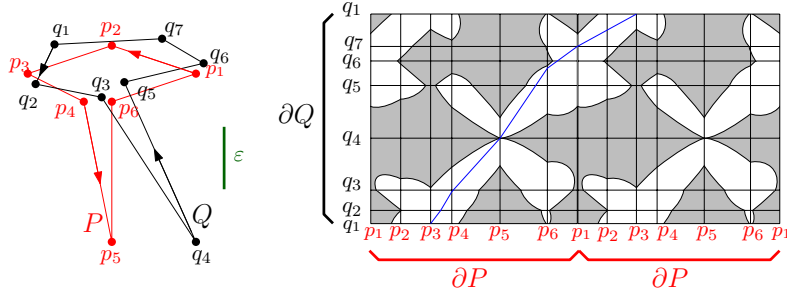


Fig. 1. The white areas are those in free space diagram. The surfaces are within Fréchet distance ε since there is a monotone path starting at the bottom of the free space diagram and ending at the top which maps every point on the boundary of P exactly once. This figure was generated using an ipelet created by Günter Rote.

Lemma 1. *The Fréchet distance between a folded polygon P and a convex polygon Q is the same as that between their boundary curves.*

As mentioned in Section 2.1, for the simple polygon algorithm it suffices to map diagonals onto shortest paths between two points on ∂Q . By contrast, there are folded polygons where a homeomorphism between the surfaces does not exist when diagonals are mapped to shortest paths but does exist when the paths are not restricted, see Figure 2. The curve s_1 is the shortest path between the points a and b but the curve s_2 has smaller Fréchet distance to d than s_1 has. We must therefore consider mapping the diagonals to more general paths. Fortunately, we can show that these more general paths still have some nice properties for folded polygons.

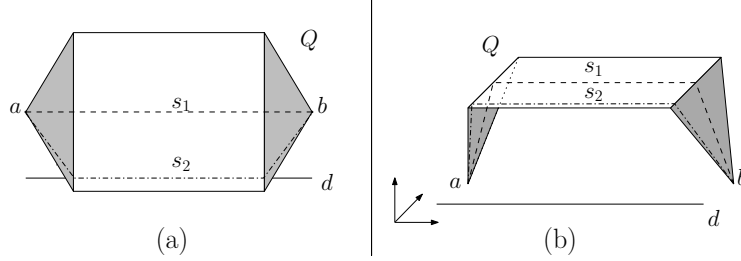


Fig. 2. An example where the image curve s_2 with the smallest Fréchet distance to a diagonal d is a non-shortest path s_1 in Q . (a) overhead view, (b) sideview.

Lemma 2. *Let Q be a folded polygon, u and v be points such that $u, v \in \partial Q$, $E = \{e_0, e_1, \dots, e_s\}$ be a sequence of edges in the convex subdivision of Q , and d be a line segment. Given $\varepsilon > 0$ we can find a curve f in Q that follows the edge sequence E from u to v such that $\delta_F(d, f) \leq \varepsilon$, if such a curve exists, in $O(s)$ time.*

Proof. We construct a series $F = FS_\varepsilon(e_0, d), FS_\varepsilon(e_1, d), \dots, FS_\varepsilon(e_s, d)$, of 2-dimensional free space diagrams. Any two edges e_i and e_{i+1} are on the boundary

of the same convex polygon in the convex subdivision of Q so we can assume without loss of generality that f consists of straight line segments between the edge intersections. This is similar to the shortcutting argument used to prove Lemma 3 in [Buc06]. Thus, we only need to check the points where f crosses an edge of Q . For $\delta_F(d, f) \leq \varepsilon$ to be true, the preimages of those crossing points must be monotone along d . Let $FS'_\varepsilon(e_i, d)$ be the projection of $FS_\varepsilon(e_i, d)$ onto d . Let $F' = FS'_\varepsilon(e_1, d), FS'_\varepsilon(e_2, d), \dots, FS'_\varepsilon(e_s, d)$.

To verify that the preimage points on d can be chosen such that they are monotone, we check the intervals of F' . Specifically, for $i < j$, the point on d mapped to e_i must come before the one mapped to e_j . This can be checked by greedily scanning left to right and always choosing the smallest point on d which can be mapped to some edge. A search of this form takes $O(s)$ time. \square

The dual graph of the faces of Q is acyclic. This implies that there is a unique sequence of faces through which a shortest path from u to v , where $u, v \in \partial Q$, must pass. Necessarily, there must also be a unique edge sequence that the shortest path follows. We call such an edge sequence the *shortest path edge sequence*.

Lemma 3. *Let d be a diagonal. If there is a curve $f \subseteq Q$ with $\delta_F(d, f) \leq \varepsilon$ then there is a curve $g \subseteq Q$ which follows the shortest path edge sequence such that $\delta_F(d, g) \leq \varepsilon$.*

Proof. Let E_f and E_g be the edge sequences of f and g respectively. By definition the dual graph of the faces of Q is acyclic, so E_g must be a subsequence of E_f . E_g induces a sequence of free space intervals. If there is a monotone path in the free space interval sequence induced by E_f , we can cut out some intervals and have a monotone path in the free space for E_g . \square

From Lemma 3, we just need to consider paths that follow the shortest path edge sequence. We refer to paths that follow this edge sequence and consist of straight line segments between edges as *Fréchet shortest paths*. In addition, s in Lemma 2 is bounded by the number of edges in along the shortest path edge sequence between u and v , and E will be the shortest path sequence. This implies the following theorem.

Theorem 1. *Let Q be a folded polygon, u and v be points such that $u, v \in \partial Q$, and d be a line segment. Given $\varepsilon > 0$, we can in $O(l)$ time find a curve f in Q from u to v such that $\delta_F(d, f) \leq \varepsilon$ if such a curve exists.*

Suppose we have a homeomorphism between ∂P and ∂Q . The endpoints of the image curves must appear on ∂Q in the same order as their respective diagonal endpoints on ∂P . The homeomorphism also induces a direction on the diagonals in P and on the edges in Q . Specifically, we consider diagonals and edges to start at their first endpoint along ∂P or ∂Q respectively in a counterclockwise traversal of the boundaries. We denote by D_e the set of diagonals whose associated shortest path edge sequences contain an edge $e \subseteq Q$. Observe that pairwise non-crossing image curves must intersect an edge e in the same order as their endpoints occur on ∂Q . We refer to this as the *proper intersection order* for an edge e .

2.3 Diagonal Monotonicity Test and Untangleability

We now define a test between two folded polygons P and Q which we call the *diagonal monotonicity test*. For a given ε this test returns true if the following two things are true. First, $\delta_F(\partial P, \partial Q) \leq \varepsilon$. Second, for every diagonal d_i in the convex subdivision of P , the corresponding Fréchet shortest path f_i in Q has $\delta_F(d_i, f_i) \leq \varepsilon$. We refer to the class of mappings of the folded polygons generated by this test as *monotone diagonal mappings*. This is similar to the test used by [Buc06] except ours uses Fréchet shortest paths instead of the shortest paths.

Unfortunately, because the image curves of the diagonals are no longer shortest paths, they may cross each other and we will no longer be able to generate a mapping between the folded polygons which is a homeomorphism. For an example of this see Appendix B. Thus the diagonal monotonicity test might return true when in fact a homeomorphism does not exist. We must explicitly ensure that the image curves of all diagonals are non-crossing. In particular, we refer to a set of image curves $F = \{f_1 \dots f_k\}$ as *untangleable* for ε if and only if there exists a set of image curves $F' = \{f'_1 \dots f'_k\}$ where f_i and f'_i have the same end points on ∂Q , $\delta_F(d_i, f'_i) \leq \varepsilon$, and the curves of F' are pairwise non-crossing. A homeomorphism exists between the folded polygons for ε if and only if there exists a monotone diagonal mapping whose image curves are untangleable for ε . This follows from the same argument used in [Buc06] for simple polygons.

As shown in Theorem 1, computing Fréchet shortest paths instead of shortest paths does not increase the asymptotic run time. To optimize this ε we can perform a binary search on a set of critical values. As in [Buc06], the number of critical values is $O(m^2n + mn^2)$. The three types of critical values between a diagonal and its corresponding path through Q are very similar to those outlined in the simple polygons algorithm. So, by following the paradigm set forth by [Buc06], we arrive at the following theorem:

Theorem 2. *The minimum ε for which two folded polygons P and Q , pass the diagonal monotonicity test can be computed in time $O(kT_{matrixmult}(mn) \log(mn))$.*

3 Fixed-Parameter Tractable Algorithm

In this section we outline an algorithm to decide for a fixed mapping between the boundaries of a pair of folded polygons whether the image curves induced from the mapping are untangleable. From this we create a fixed-parameter tractable algorithm for computing the Fréchet distance between a pair of folded polygons.

3.1 Untangleability Space

Let e be an edge in Q which is crossed by the image curves of h diagonals, d_1, \dots, d_h . We assume without loss of generality that the image curves of the diagonals cross e in proper intersection order if, for all $1 \leq i, j \leq h$ where $i < j$, the image curve of d_i crosses e before the image curve of d_j crosses e . Let the *untangleability space* U_e contain all k -tuples of points on the diagonals which can

be mapped to crossing points on the edge e within distance ε and such that the crossing points are in the proper intersection order along e . U_e can be shown to be convex yielding the following theorem.

Theorem 3. $U_e(d_1, \dots, d_k)$ is convex.

This theorem can be proven by linearly interpolating between points in U_e . For the full proof see Appendix C.

3.2 Fixed-Parameter Tractable Algorithm

We assume the complexity of k and l , the convex subdivisions of P and Q are constant. Checking for the existence of a set of image curves which are untangleable can be done by using the untangleability spaces of the diagonals. Assume we are given some homeomorphism between ∂P and ∂Q from which we get a placement of the diagonals. We first choose an edge in Q to act as the root of the *edge tree* that corresponds to the dual graph of Q . We propagate constraints imposed by each untangleability space up the tree to the root node to determine if the set of image curves induced by the placement of the diagonals is untangleable.

The untangleability space of an edge e , U_e , contains exactly those sets of points on the diagonals in D_e which can be mapped to the edge in the proper intersection order. The point chosen in U_e imposes a constraint on what points may be chosen in other untangleability spaces. In particular the corresponding points on all of the diagonals must be monotone with respect to their edge sequence. We define $C(U_{e_i})$ as the Minkowski sum of U_{e_i} with a ray in the opposite direction of the constraint on each of the diagonals in D_{e_i} , see Figure 3. In Figure 3 U_e is shown in white and $C(U_e)$ is the union of the white and light gray portions. The direction of this constraint depends on which side of the edge e_i the next edge is. $C(U_{e_i})$ contains exactly those sets of points on the diagonals not excluded from having a monotone mapping with U_{e_i} .

We define for every edge e a k -dimensional *propagation space* P_e . If e is a leaf in the tree, then $P_e = U_e$. Otherwise, define

$$P_e = U_e \cap C(P_{e_1}) \cap \dots \cap C(P_{e_j})$$

where e is the parent of the edges e_1, e_2, \dots, e_j . $C(P_{e_j})$ contains only those points that are not excluded by the constraints of the tree rooted at e_j from being used to untangle on the parent of e_j . The propagation space for the root will be empty if and only if this set of image curves are not untangleable. From our assumptions, the propagation space of the root can be computed in constant time as the intersection of semi-algebraic sets [Bas06]. Let $F(k, l)$ be the time complexity of computing this intersection.

Consider two different mappings between ∂P and ∂Q . These determine different placements of the diagonals. If all of the image curves of all of the diagonals have the same shortest path edge sequence in both of the mappings the test

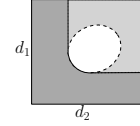


Fig. 3.

will return the same result. Thus, we only need to test paths through the free space diagram which cross the diagonals and edges in a different order. The free space diagram for ∂P and ∂Q contains $2k$ vertical line segments that will each contribute $O(kl)$ different mappings of the diagonals and edges. Hence, there are $O((kl)^{2k})$ paths through the free space diagram which we need to test. For each of these we can check whether a global untangling exists as described above in constant time. Similar to the algorithm for polygonal curves [Alt95] we can perform Cole's [Col87] technique for parametric search [Meg83] to optimize the value of ε . For convex subdivisions with constant number of edges l and diagonals k , this yields a fixed-parameter tractable algorithm with runtime polynomial in m and n .

Theorem 4. *We can compute the Fréchet distance of two folded polygons in time $O((F(k, l)(kl)^{2k} + kT_{matrixmult}(mn)) \log(mn))$.*

4 Constant Factor Approximation Algorithm

In this section we present an approximation algorithm based on our diagonal monotonicity test to avoid tangles altogether. First we demonstrate the following theorem:

Theorem 5. *If two folded polygons, P and Q , pass the diagonal monotonicity test for some ε , then $\delta_F(P, Q) \leq 9\varepsilon$.*

Proof. Consider the image curves of the diagonals of P found by performing our diagonal monotonicity test. To pass the diagonal monotonicity test there must be a homeomorphism between ∂P and ∂Q . The image curves of the diagonals will be mapped in the proper order along the boundary of Q . Therefore, if a pair of image curves cross in Q they must do so an even number of times.

Take two consecutive points u, v on the boundary of P that are connected by a diagonal $d = \overline{uv}$, and consider the convex ear of P that d "cuts off". Let d' be the image curve of d in Q . In order to create a homeomorphism between P and Q , d' should cut off an ear of Q which can be mapped to the ear of P . Unfortunately, some image curves may cross this d' and cause tangles. Consider the arrangement of image curves in Q and let d'' be the highest level of this arrangement closest to the top of the ear (∂Q), such that d'' connects the image points u and v on the boundary of Q .

Observe that if an image curve d'_1 crosses d' from below then that intersection point a' has a pre-image on both d and d_1 . These points a on d and a_1 on d_1 can be no more than 2ε apart since they both map to the intersection within distance epsilon. In addition, d'_1 must cross back below d' eventually since all image curves which cross do so an even number of times (take the first such occurrence after the initial crossing). The preimage points b and b_1 of this second intersection b are also no more than 2ε apart. Since both d and d' are line segments, every point on the line segment \overline{ab} on d is $\leq 2\varepsilon$ distance from some point on the line segment $\overline{a_1b_1}$ on d_1 . For an approximation of 3ε we can map a point on d to its

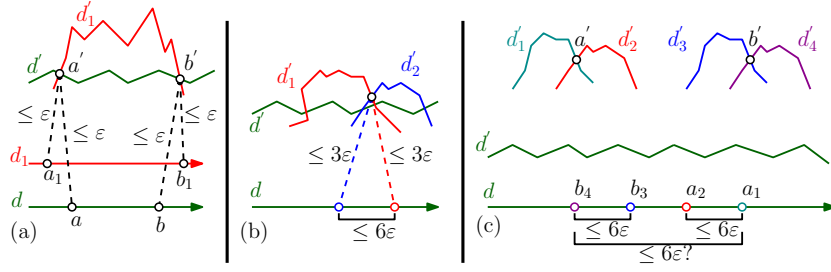


Fig. 4. (a) the intersections between d and d_1 imply that \overline{ab} can be mapped to the region of d'_1 between a' and b' within Fréchet distance ε (b) this is the preimage of the intersection between d'_1 and d'_2 on d , (c) an example of the preimages of two intersections occurring out of order of d

corresponding point on d_1 and then to where that point maps on d'_1 , see Figure 4(a). Hence, the diagonal d can be mapped to an image curve on or above d'_1 within Fréchet distance 3ε .

If this image curve d'_1 then crosses another image curve d'_2 this argument above cannot be just repeated because the approximation factor would depend linearly on the number of image curves which cross each other. The preimage points on d of such an intersection not involving d' are separated by at most 6ε . This is because both diagonals involved in the intersection have a 3ε correspondence between the region of them mapped above d' and d . If the preimages are in order there is no problem. If they occur out of order they cause a monotonicity constraint. Fortunately, we can collapse this region on d to the leftmost preimage with 6ε and then map it to the corresponding point on d'_2 in 3ε for a total of 9ε , see Figure 4(b). If the preimage points in this example were reversed they would be in order.

Thus, we can approximate away the monotonicity constraint of single intersections with 9ε . We must also verify that if the preimages of single intersections occur out of order it does not effect our approximation, see Figure 4(c). Due to lack of space the discussion of these technical cases has been moved to the appendix, see Appendix D.

From these we get that $\delta_F(d, d'') \leq 9\varepsilon$. Now collapse the ear we initially selected in P to d . Likewise in Q collapse the corresponding ear to d'' . This is okay because d'' is above all of the other image curves in Q . This pairs the ear we cut off of P with the part of Q above d'' which is a folded polygon (it could be simpler than a folded polygon but we know it's no more complex than that). By Lemma 1 the ear of P and the folded polygon above d'' must be within Fréchet distance 9ε .

Choose another ear in P . We can repeat the above arguments to remove this new ear and its corresponding ear in Q . The dual graph of P is a tree. Each time we repeat this argument we are removing a leaf from the tree. Eventually, the tree will contain only a single node which corresponds to some triangle in P which we map to the remainder of Q . \square

As a direct consequence of Theorems 5 and 2 we get the following theorem:

Theorem 6. *We can compute a 9-approximation of the Fréchet distance of two folded polygons in time $O(kT_{\text{matrixmult}}(mn) \log(mn))$.*

5 Axis-Parallel Folds and L_∞ Distance

The counter example from Appendix B works for all L_p except for L_∞ . In this section we outline a special case where using the L_∞ metric guarantees that if a pair of folded polygons pass the diagonal monotonicity test for ε their Fréchet distance is no more than ε . Specifically, if all of the line segments in the convex subdivision of the surfaces are parallel to the x-axis, y-axis, or z-axis, we show that it is sufficient to use shortest paths instead of Fréchet shortest paths. Since shortest paths never cross we can use the simple polygons algorithm in [Buc06] to compute the Fréchet distance of the surfaces. We first prove the following lemma.

Lemma 4. *Let R be a half-space such that the plane bounding it, ∂R , is parallel to the xy -plane, yz -plane, or xz -plane. Given a folded polygon Q with edges parallel or perpendicular to the x -axis and points $a, b \in Q \cap R$, let f be a path in Q , which follows the shortest path edge sequence between a and b . If P is completely inside of R so is the shortest path f' between a and b .*

For the lemma to be false there must exist a Q , R , and f which serve as a counter example. There must be at least one edge e_j in Q such that $f \cap e_j \in R$ and $f' \cap e_j \notin R$. In particular, let e_j be the first edge where this occurs along the shortest path edge sequence. First consider a Q where all of the edges of it are perpendicular to ∂R . A line segment in the shortest path f' connects the endpoints of two edges in Q . Let e_i and e_k be the edges that define the line segment in f' that passes through e_j . We now consider several cases in how those edges are positioned.

Case (I) occurs when e_k is completely outside of R , see Figure 5(a). While this does force f' to cross e_j outside of R , there is no f which can pass through e_k while remaining inside of R . Because Q is a folded polygon any path between a and b must path through the edges in the shortest path edge sequence including e_k . Thus no f can exist entirely within R .

Case (II) occurs when part of e_k is in R and f' crosses it in the part in R , see Figure 5(b). In this case f' does not cross e_j outside of R .

Due to space limitations the discussion of the remaining cases has been moved to the appendix, see Appendix E. Each of the remaining cases can be reduced to these first two. Using this lemma we can prove the following theorem:

Theorem 7. *The Fréchet distance between two surfaces, both with only diagonals/edges parallel to the x -axis, y -axis, or z -axis, can be computed in time $O(kT_{\text{matrixmult}}(mn) \log(mn))$.*

Proof. Let Q be a folded polygon, d be a diagonal, and f' be the shortest path between points a and c on ∂Q . Using Lemma 4 we prove that if there exists a Fréchet shortest path f between points a and c such that $\delta_F(d, f) \leq \varepsilon$, then $\delta_F(d, f') \leq \varepsilon$.

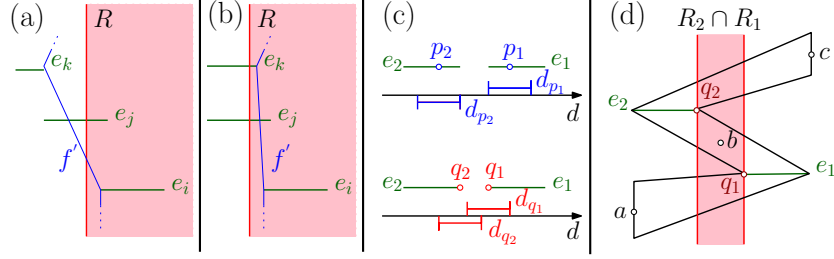


Fig. 5. (a), (b) are examples of case (I) and case (II). (c) example intervals for the two different paths. (d) together the edges e_1 and e_2 cause a monotonicity constraint.

Minkowski Sum Constraints Since we are using the L_∞ distance, the unit ball is a cube with sides of length 1. The Minkowski sum of a diagonal d in P and a cube of side length ε yields a box. Points in the diagonal d can only map to points in this region. It can be defined by the intersection of 6 half-spaces; all of these have boundaries parallel to either the xy -axis, the xz -axis, or the yz -axis. Thus, from Lemma 4, we know that if any path through Q is completely within this box, then the shortest path f' will be, too. This means that for each edge e_i on the shortest path edge sequence $f' \cap e_i$ is within distance ε of some non-empty interval of d .

Monotonicity Constraints For the shortest path f' between the boundary points to have $\delta_F(d, f') > \varepsilon$, at least two of these intervals must be disjoint and occur out of order along d , see Figure 5(c). Such a case introduces a monotonicity constraint on ε . If no such intervals existed then we could choose a monotone sequence of points along d such that each point is within distance ε of an edge and the sequence of edges they map to would have the same order as the shortest path edge sequence showing that $\delta_F(d, f') \leq \varepsilon$.

Let e_1 and e_2 be two edges along the shortest path edge sequence for which such bad intervals occur. Let p_1 and p_2 be points on the shortest path where it intersect edges e_1 and e_2 respectively. Let q_1 and q_2 be the same for f . Finally, let d_r contain all of the points on d which are within distance ε of the point r . Since $\delta_F(d, f) \leq \varepsilon$, d_{q_1} and d_{q_2} must overlap or occur in order along d .

Let R_1 be the half-space whose bounding plane contains q_1 and is perpendicular to d . likewise let R_2 be the half-space whose bounding plane contains q_2 and is perpendicular to d , see Figure 5(d). Let R_1 extend to the left along d and R_2 extend to the right along d . ∂R_2 must occur before ∂R_1 along d or the edges are in order and no monotonicity constraint is imposed. Assume R_1 encloses all of f' between a and e_2 . If it does not we can choose a new edge between a and e_2 to use as e_1 for which this is true. Doing so only increases the monotonicity constraint. Likewise we can assume R_2 encloses all of f' between e_2 and c .

Assume a , b , and c lie on f' . Specifically, let a and c be the end points of f' on ∂Q . Naturally, a shortest path must exist between a and c and it must contain at least one point in $R_1 \cap R_2$ which we call b . f follows the shortest path edge sequence between a and c , so it must also cross all of the edges in the shortest

path edge sequence between a and b . Therefore, to show that p_2 is inside of R_1 we can directly apply Lemma 4 to the points a and b . A similar method can be used for e_2 with points b and c to show p_1 is inside R_2 . Since d_{q_1} and d_{q_2} overlap or are in order, d_{p_1} and d_{p_2} must as well. Therefore, $\delta_F(d, f') \leq \varepsilon$ and shortest paths can be used for this variant of folded polygons instead of Fréchet shortest paths. Because we are using shortest paths we can just use the simple polygons algorithm. This yields Theorem 7. \square

6 Future Work

The constant factor approximation outlined in Section 4 can likely still be improved. Specifically, we consider only the worst case for each of the out-of-order mappings which may not be geometrically possible to realize. In addition, we currently approximate the Fréchet distance by mapping image curves one-by-one to the top of the arrangement of other image curves. It would of course be more efficient to untangle image curves by mapping them to some middle curve rather than forcing one to map completely above the others.

Finally, while the problem of untangling seems hard, it is also possible that a polynomial-time exact algorithm could exist. The acyclic nature of our surfaces seems to limit the complexity of our mappings. The methods used to prove that computing the Fréchet distance between certain classes of surfaces is NP-hard in [Buc10] are not easy to apply to folded polygons.

References

- Alt95. H. Alt and M. Godau. Computing the Fréchet distance between two polygonal curves. *International Journal of Computational Geometry and Applications*, 5:75–91, 1995.
- Alt10. H. Alt and M. Buchin. Can we compute the similarity between surfaces? *Discrete and Computational Geometry*, 43:78–99, 2010.
- Bas06. S. Basu, R. Pollack, and M.-F. Roy. Algorithms in real algebraic geometry. *Algorithms and Computation in Mathematics*, 2006.
- Buc06. K. Buchin, M. Buchin, and C. Wenk. Computing the Fréchet distance between simple polygons in polynomial time. *22nd Symposium on Computational Geometry (SoCG)*, pp. 80–87, 2006.
- Buc10. Kevin Buchin, Maike Buchin, and André Schulz. Fréchet distance of surfaces: Some simple hard cases. In *Proc. 18th Annual European Symposium on Algorithms (ESA), part II*, vol. 6347, pp. 63–74. Springer, 2010.
- Col87. R. Cole. Slowing down sorting networks to obtain faster sorting algorithms. *J. ACM*, 34(1):200–208, 1987.
- God98. M. Godau. *On the complexity of measuring the similarity between geometric objects in higher dimensions*. Ph.D. thesis, Freie Universität Berlin, Germany, 1998.
- Meg83. N. Megiddo. Applying parallel computation algorithms in the design of serial algorithms. *J. ACM*, 30:852–865, 1983.

A Proof of Lemma 1

Proof. Assume that the Fréchet distance between the boundary curves is ε . Now consider mapping the diagonals in the convex decomposition of P to Q . Because Q is convex, we can connect the images of the endpoints of each of the diagonals with a line segment. These line segments subdivide Q into a set of convex polygons each of which are paired with a convex polygon in P . The boundaries of the paired polygons must be within Fréchet distance ε of each other, consequently, by [Buc06], the paired convex polygons must be in Fréchet distance ε .

Combining all of these mappings, we have a homeomorphism between P and Q . Thus the Fréchet distance of P and Q must also be $\leq \varepsilon$. \square

B Counter Example

In this section we show a counter example to the claim that it is sufficient that a pair of folded polygons pass the diagonal monotonicity test for ε to guarantee that they are within in Fréchet distance ε . Let P and Q be the surfaces shown in Figures 6(a) and 6(b) with $\varepsilon = 1$.

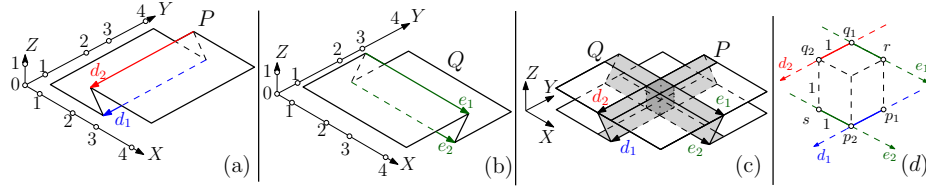


Fig. 6. (a),(b) The folded polygons P and Q , (c) P and Q overlaid such that the zigzag portions of the surfaces outline a cube at their center, (d) The cube has sides of length one. The marked points must be used in the mappings for $\varepsilon = 1$ but they can be shown to be mutually exclusive

The surfaces are the same except rotated by 90 degrees. Each surface consists of two parallel 2 by 3 rectangles which are connected together by a smaller slanted rectangle. One of the large rectangles is at height 1 while the other is at height 0. Figure 6(c) shows how the two surfaces are overlaid.

We can prove that the boundary curves have Fréchet distance 1 to each other. Pair up each side of P with one in Q as indicated in Figure 7(b). These paired portions of the boundary have Fréchet distance 1 to each other. The case of mapping A to A' is given in Figure 7(c). For the beginning and ending of the sides the curves either overlap or are exactly distance 1 apart with one directly above the other, so the mapping used is obvious. Matching the flat middle part on A' to the zigzag in A is a bit harder. The key insight is to observe that the

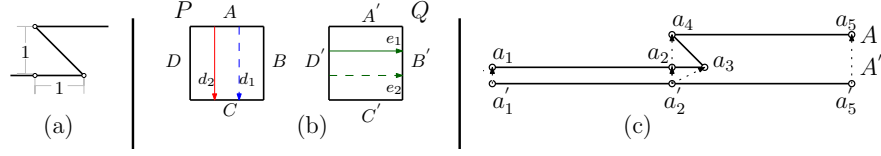


Fig. 7. (a) Sideview of the surfaces, (b) Top view of P and Q showing which sides are mapped to which, (c) A side of P is mapped onto a side of Q . a_1 and a'_1 as well as a_2 and a'_2 occupy the same points in space and a separation is shown for clarity.

point a'_2 is within distance 1 to a_2 , a_3 , and a_4 . The entire zigzag can be mapped to this point. The portion after the point a'_4 can then be mapped straight down.

Consider the intersection between P and Q . At the center of the folded polygons is a cube with side length 1, see Figures 6(c) and 6(d). The point p_1 on d_1 can map both to the point p_2 on edge e_2 and to the point r on edge e_1 within distance 1. Thus, there exists an image curve f_1 such that $\delta_F(d_1, f_1) \leq 1$. Likewise, the point q_2 on d_2 can map both to the point s on edge e_2 and to the point q_1 on edge e_1 within distance 1, so there exists an image curve f_2 such that $\delta_F(d_2, f_2) \leq 1$. Therefore, P and Q pass the diagonal monotonicity test for 1.

Notice that p_1 is the only point on d_1 which can map to e_1 . Given the direction of d_1 , f_1 starts at the top of Q and must cross edge e_2 then e_1 . Thus no points after p_1 on d can be mapped to e_2 so p_2 is the only reachable point on e_2 . Likewise, s is the only point on e_2 the diagonal d_2 can map to. s comes before p_2 along e_2 so the image curve of d_2 must cross e_2 before the image curve of d_1 . This is different from the proper intersection order that comes from the boundary. Therefore, even though P and Q passed the diagonal monotonicity test for 1, $\delta_F(P, Q) > 1$.

C Convexity of the Untangleability Space

Proof. To construct a proof we first define U_e more formally.

$$U_e(d_1, \dots, d_h) = \{a = (a_1, \dots, a_h) | (\exists p_1, \dots, p_h \in e | (\forall i, j) (a_i \in d_i, 1 \leq i \leq j \leq h) \rightarrow (\|a_i - p_i\| \leq \varepsilon, p_i < p_j)))\}$$

where $p_i \leq p_j$ means that the point p_i is no further than the point p_j along the edge e . If U_e is not convex then there are points $a = (a_1, \dots, a_h), c = (c_1, \dots, c_h) \in U_e$ such that there exists a point $b = (b_1, \dots, b_h)$ on the line segment between a and c where $b \notin U_e$, see Figure 8a. The point b is on the line segment between a and c so there exists an x between 0 and 1 such that $b = xa + (1 - x)c$. Likewise for each point $b_i \in b$, $b_i = xa_i + (1 - x)c_i$. Let p_i and r_i be the points on edge e mapped to by points a_i and b_i respectively. Let

$q_i = xp_i + (1-x)r_i$. Since $a, c \in U_e$, $\|a_i - p_i\| \leq \varepsilon$ and $\|c_i - r_i\| \leq \varepsilon$. It follows from these that $\|b_i - q_i\| \leq \varepsilon$, see Figure 8b.

$$\begin{aligned}\|a_i - p_i\| \leq \varepsilon &\Rightarrow x\|a_i - p_i\| \leq x\varepsilon \\ &\Rightarrow \|xa_i - xp_i\| \leq x\varepsilon\end{aligned}$$

$$\begin{aligned}\|c_i - r_i\| \leq \varepsilon &\Rightarrow (1-x)\|c_i - r_i\| \leq (1-x)\varepsilon \\ &\Rightarrow \|(1-x)c_i - (1-x)r_i\| \leq (1-x)\varepsilon \\ &\Rightarrow \|(1-x)c_i - (1-x)r_i\| \|xa_i - xp_i\| \leq (1-x)\varepsilon + x\varepsilon \\ &\Rightarrow \|(xa_i + (1-x)c_i) - (xp_i + (1-x)r_i)\| \leq \varepsilon - x\varepsilon + x\varepsilon \\ &\Rightarrow \|b_i - q_i\| \leq \varepsilon\end{aligned}$$

Since $a, c \in U_e$, $p_i \leq p_j$ and $r_i \leq r_j$. It follows from a proof similar to the previous one that $q_i \leq q_j$, see Figure 8c.

$$p_i \leq p_j \Rightarrow xp_i \leq xp_j$$

$$\begin{aligned}r_i \leq r_j &\Rightarrow r_i \leq r_j \\ &\Rightarrow (1-x)r_i \leq (1-x)r_j \\ &\Rightarrow xp_i + (1-x)r_i \leq xp_j + (1-x)r_j \\ &\Rightarrow q_i \leq q_j\end{aligned}$$

Thus, for all i, j , where $0 \leq i \leq j \leq h$, $\|b_i - q_i\| \leq \varepsilon$ and $q_i \leq q_j$. From this it follows by definition that $b \in U_e$. Therefore U_e is convex. \square

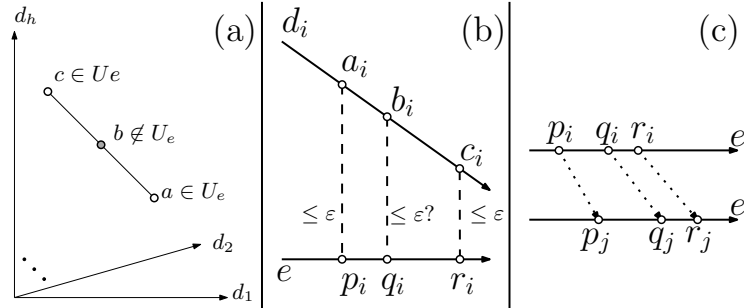


Fig. 8. (a) $a, c \in U_e$ but $b \notin U_e$. (b) both a_i and c_i being within distance ε of their respective points on e necessitates that b_i is within distance ε of its own. (c) The crossing points of diagonals d_i and d_j being in order along e for both points a and c necessitates that they are also in order for b

D Constant Factor Approximation Order of Regions

We now need to show that the points that the intersections are mapped to do not occur out of order. This could happen when the preimage regions on d occur out of order, see Figure 4(c). In this case, collapsing to the leftmost point of a region for a single intersection would not be sufficient. The two collapsed points would still be out of order along d .

We consider two cases. For each of these cases we assume that the intersection points a and b are at on the top level of the arrangement of the image curves.

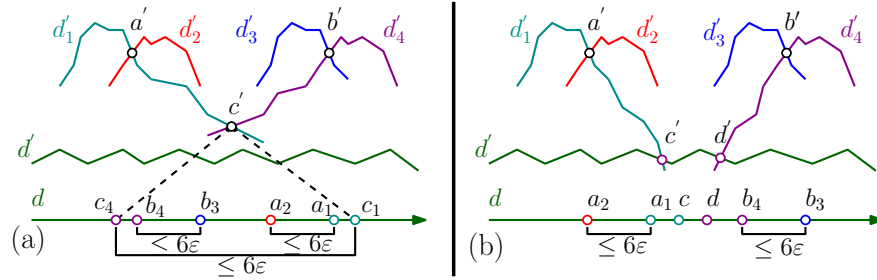


Fig. 9. (a) If the image curves intersect before crossing d' , the preimages of both intersections must completely occur on a region of d smaller than 6ε , (b) If the image curves intersect d' before each other, the preimages of the intersections must occur in the proper order

Case I

See figure Figure 9(a). Consider the image curve of the first intersection which is sloping toward d' (d'_1 in example) and the image curve of the second intersection which is sloping away from d' (d'_4 in example) intersect each other before intersecting d' (moving left-to-right). Assume d'_1 first intersects d'_2 and then d'_4 , and assume d'_4 first intersects d'_2 and then d'_3 .

In this case we still have a correspondence between the two diagonals and d (they are still above d and thus a mapping exists). The preimages of the intersection points on d are restricted to being outside the regions from Figure 4(c). The region they enclose must be $\leq 6\varepsilon$ in width. Thus we can just map the whole region to the leftmost point.

Case II

See Figure 9(b). The image curves cross d' before they intersect each other. The order the curves intersect in places a restriction on how far back the preimage of intersection 2 can be. Specifically, in this case, the preimages of the two intersections must occur in order on d so there is no new restriction imposed by them.

Induced Mapping

One of these two cases must occur. In particular, the case where the image curves cross earlier or later than specified in Case I cannot happen. In this hypothetical case one of the image curves would be forced above the other one. Assume d'_1 is the one forced above. In that case d'_1 would need to pass above the intersection b' in order to not cross d'_4 . We assumed earlier that b' was at the highest level of the arrangement so this is a contradiction. The same argument works if d'_4 is the image curve forced above.

From these two cases we can see that monotonicity constraints will not occur between different intersection preimages on d . Thus, we can collapse each of them to a single point using the 9ε approximation described above without the collapsed points occurring out of order.

E Remaining Cases for Lemma 4

Case (III) occurs when part of e_k is in R and f' crosses it in the part outside of R , see Figure 10(a). The path f can cross the part of e_k inside R . This case can be repeated many times but eventually, f' ends at b so it needs to cross back into R . To do this it must cross an edge on the opposite end point. If that end point is in R the entire path can be shortcut similar to case (II) and if the point is not in R then as in case (I) there does not exist an f which can cross e_l and remain in R see Figure 10(b). Thus case (III) also leads to a contradiction.

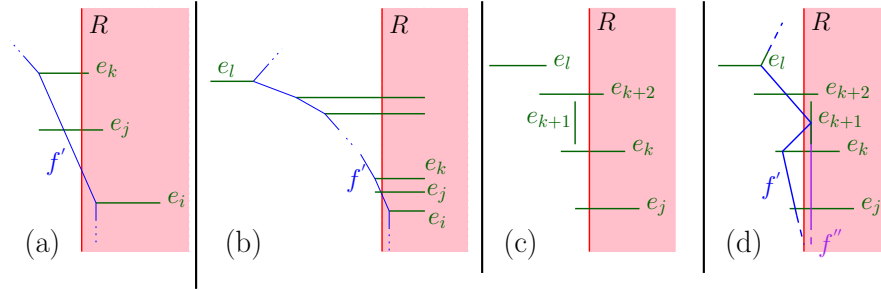


Fig. 10. (a) the edge e_{k+1} forces f' out of R on e_j , but for this to happen e_{k+1} must be completely outside of R , thus no f can cross e_{k+1} . (b) f' is not forced out of R on e_j but rather can follow the path f'' . (c) the path can continue to curve away from R . (d) in this case, while the shortest path goes out of R eventually there must be an edge e_l where it reverses direction since f' ends at b which is in R . Such an edge corresponds to case (I).

Next we consider the two cases that arise from adding a perpendicular fold e_{k+1} . Specifically, we will consider where the first perpendicular fold after e_j is

placed. Such a fold runs parallel with ∂R so it must be either entirely inside R or entirely outside.

Case (IV) occurs if the edge e_{k+1} is outside of R . In this case there does not exist an f which can pass through e_{k+1} and be entirely in R , see Figure 10(c).

Case (V) occurs if the fold e_{k+1} is inside of R . In this case the shortest path f' will no longer be forced outside of R on the edge e_{j+1} . Instead, it goes through R to the edge e_{k+1} , see Figure 10(d). The only way that f' could not pass through R would be if one of the edges between e_j and e_k were entirely outside R and that is ruled out by case (I)

Thus no such Q , R , and f can exist.